Regularised Logistic Regression

Nikhil Jangamreddy (2018csm1011)

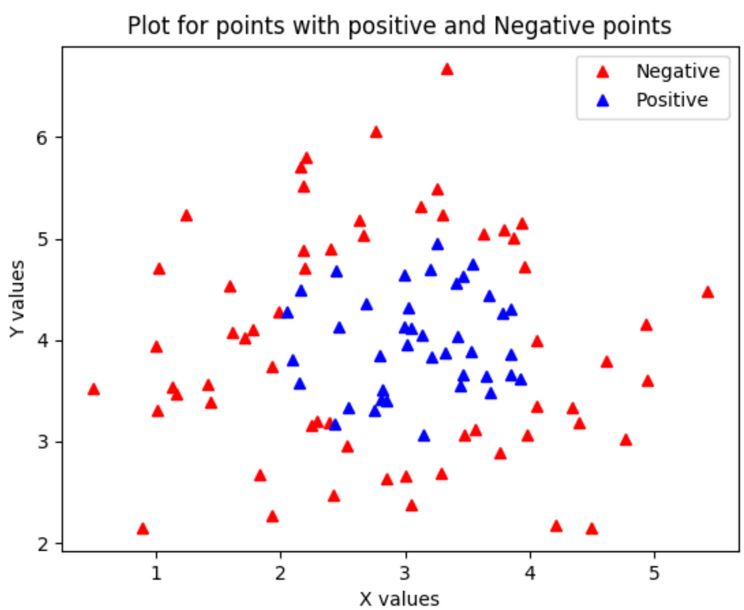
February 22, 2019

1 Regularised Logistic Regression :

In this experiment, we implement regularised logistic regression using Gradient Descent as well as Newton Raphson method. We then implement feature transformation to convert data into higher dimension space for different degree and implement logistic regression on it. We analyse performance of Logistic Regression by varying Regularisation parameter.

1.1 Plotting the Given dataset :

The given dataset corresponds to **credit.txt** file which contains 100 rows 3 columns. Out of 3 columns,first 2 columns correspond X1,X2 values and 3rd column corresponds to binary output variable associated with it.

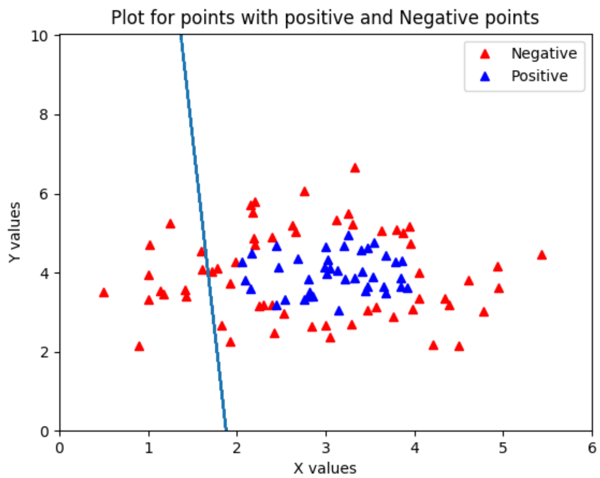


Positive Vs Negative points

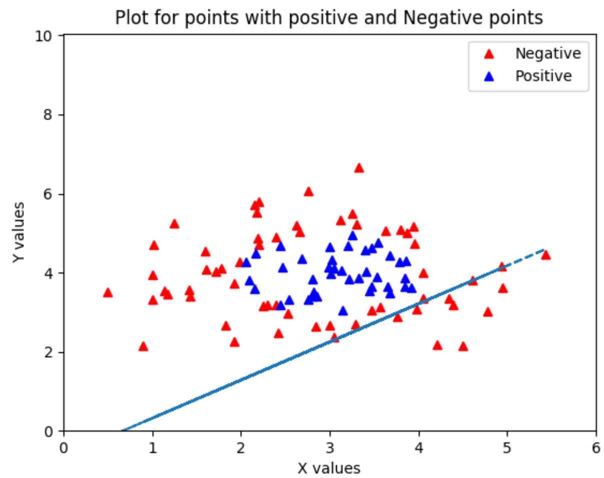
1.2 Implementing Regularised logistic regression :

Now implementing regularised logistic regression using Gradient descent method, and Newton Raphson method. Number of Iterations taken for each method is fixed. Initial weight values are choosen between [-0.1,0.1].

Parameters Used for Gradient Descent and Newton Raphson Method :
Regularisation Constant = 0.001
Learning rate = 0.0000001
Number of Iterations = 1,00,000



Plot for Gradient Descent



Plot for Newton Raphson

Error for Gradient Descent Method is **0.6776482397790687**.
Error for Newton Raphson Method is **1.958521590741952e-06**.
Note: It can be observed that Error values can change depending on the weight initialisation.

1.2.1 Observations - Gradient Descent Method :

- The rate of Convergence depends on the learning rate.
- Gradient Descent takes large number of iterations to converge, but work done in each iteration is less which leads to slower convergence per iteration.
- For slight change in learning rate, Error or Decision boundary may change drastically.

1.2.2 Observations - Newton Raphson Method :

- Convergence is faster compared to Gradient Descent Method, generally takes fewer number of iterations.
- Computing Hessian for each iteration is computationally expensive.
- Each iteration takes more time compared to time per iteration in Gradient Descent method.

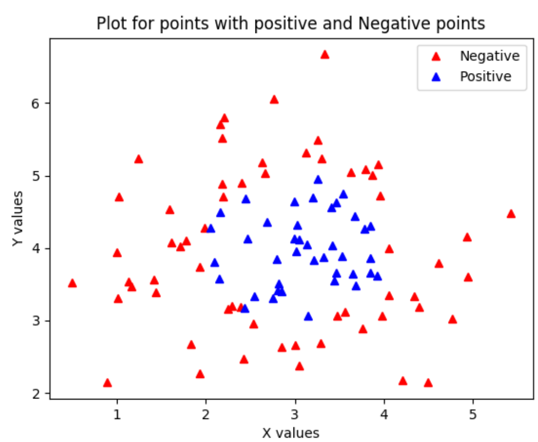
1.3 Is Data linearly Separable?

No, data is not linearly Separable as plotted in 1.1. But in below mentioned 1.4., we perform feature transformation which converts data to higher dimensions where data may be linearly separable. Although data can be linearly separable in higher dimension, it forms non linear decision boundary in two dimension.

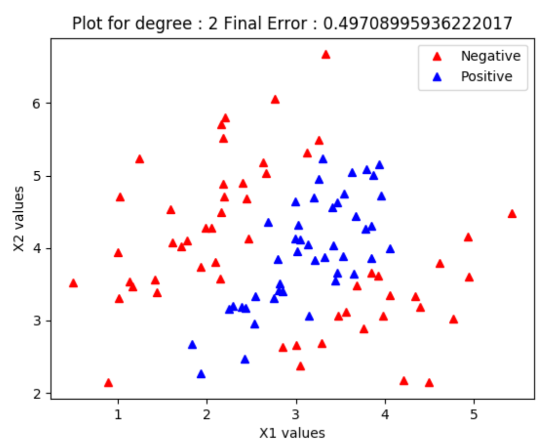
1.4 Logistic regression for varying degree values :

Now we perform feature transformation to increase the number of attributes and perform regularised logistic Regression. As specified in the Question, we took only one method for below experiments i.e.., Gradient Descent method. For Different degree, we calculate the error values along with that we plot prediction of new models for each degree.

For Degree = 2

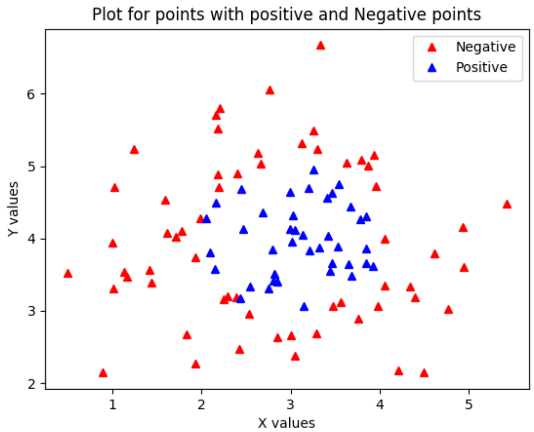


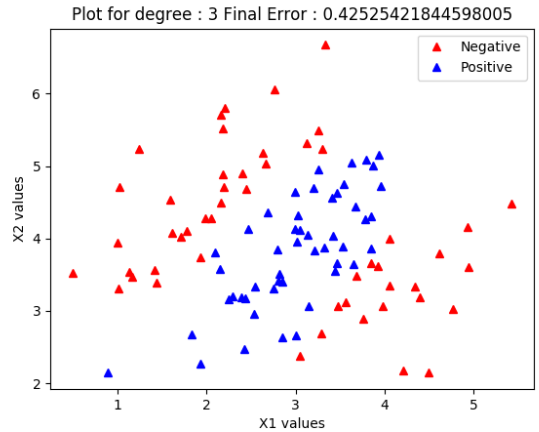
Actual values



Model prediction with degree = 2

For Degree = 3

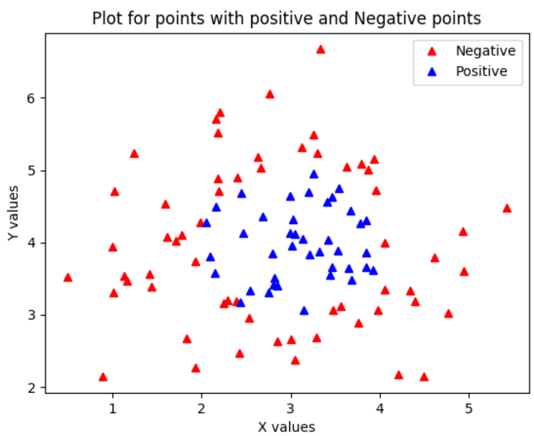


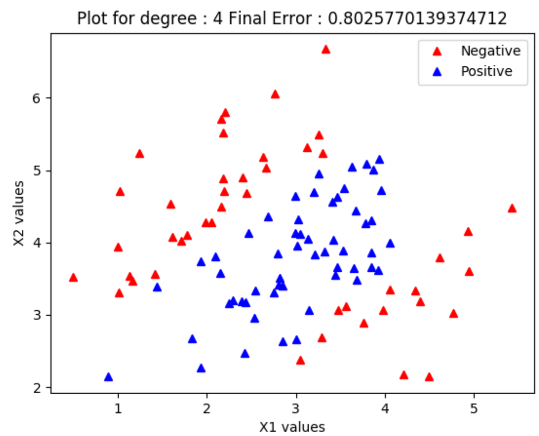


Actual values

Model prediction with degree = 3

For Degree = 4

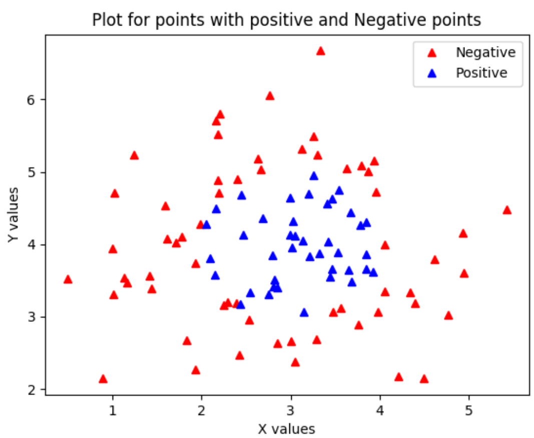


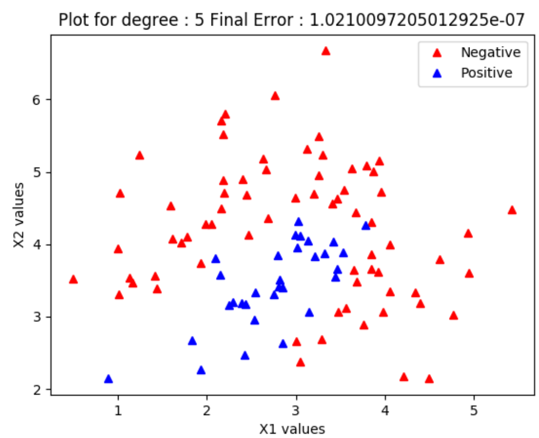


Actual values

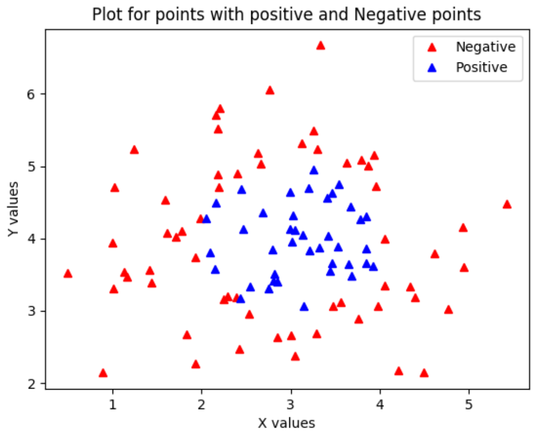
Model prediction with degree = 4

For Degree = 5

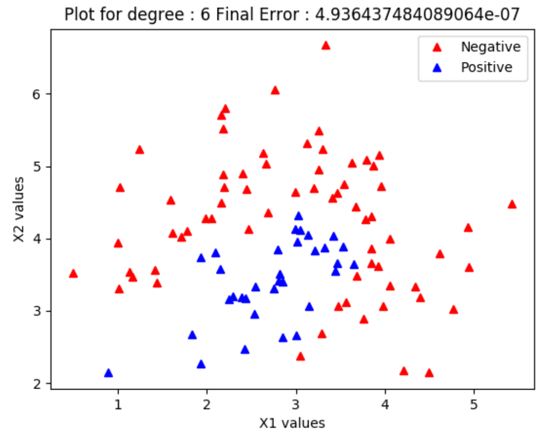




For Degree = 6

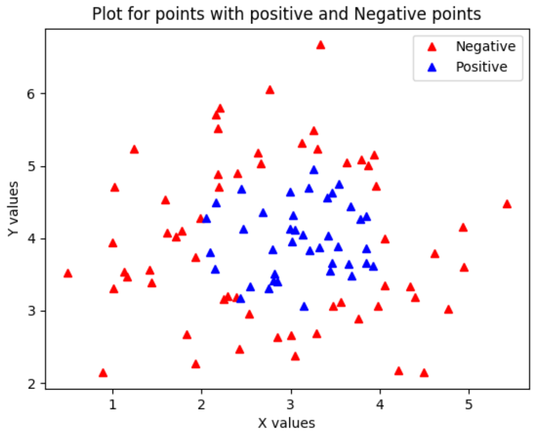


Actual values

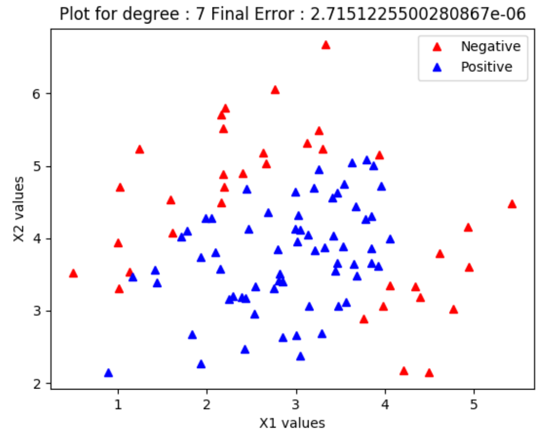


Model prediction with degree = 6

For Degree = 7

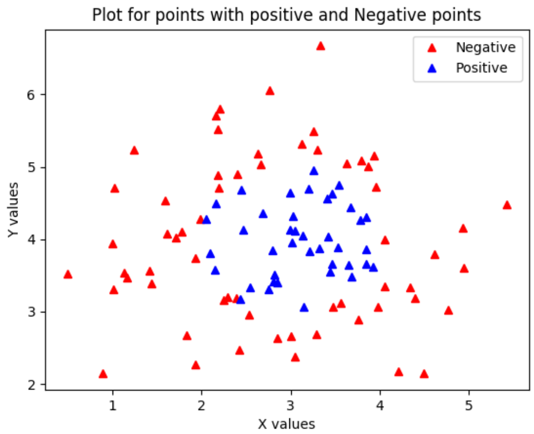


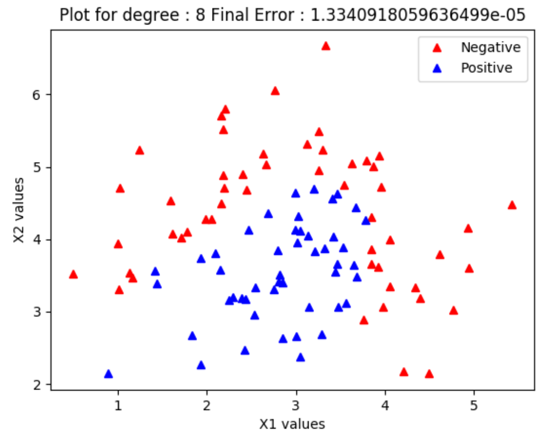
Actual values



Model prediction with degree = 7

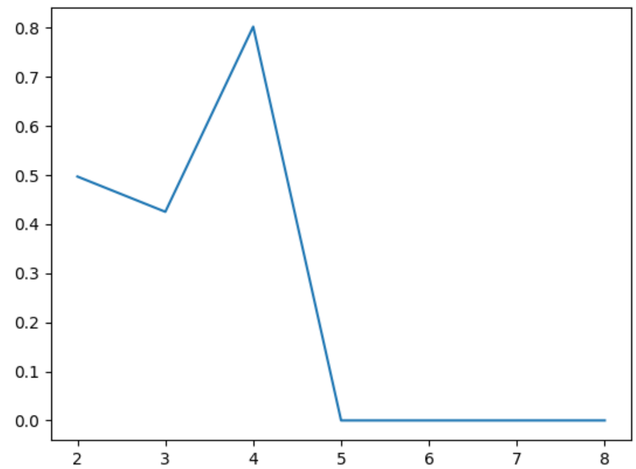
For Degree = 8





1.4.1 Error Vs Degree Graph :

As observed above predictions of model for various degrees. Now we plot Error Vs Degree to understand the model performance with respect to Degree.



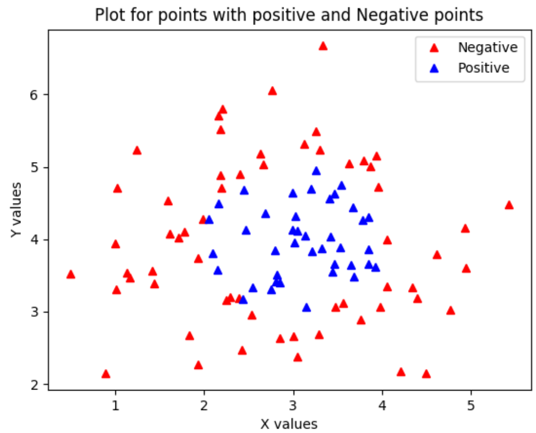
Cross Entropy error Vs Degree Values

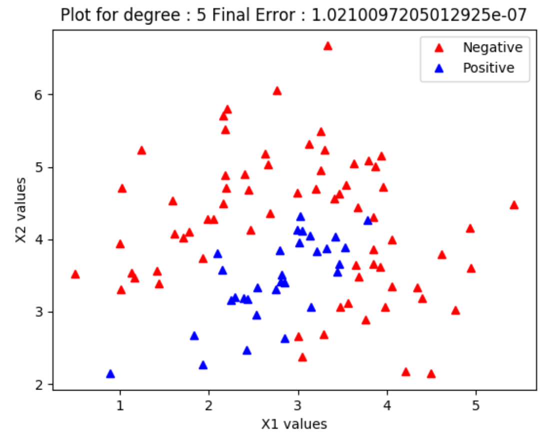
1.4.2 Observations :

- As seen from Error Vs Degree plot, Error is minimum for degree = 5. In the graph after degree 5 it may seen all errors are equal, but they are not equal. Error after degree 5 is increasing.
- From degree >=6, Error is increasing (Values are close to 0, see plots for values) which implies model is overfitting.
- Initially model is Under fitting, at degree = 5 we got best fit. After degree >=6 model is overfitting. So we choose degree = 5.

1.5 Plotting Non linear Decision Boundary :

Let us consider plot for Degree = 5, we can observe predictions of our new model predictions are non linear because if we draw a convex hull around positive predictions we get non-linear decision boundary.
For Degree = 5





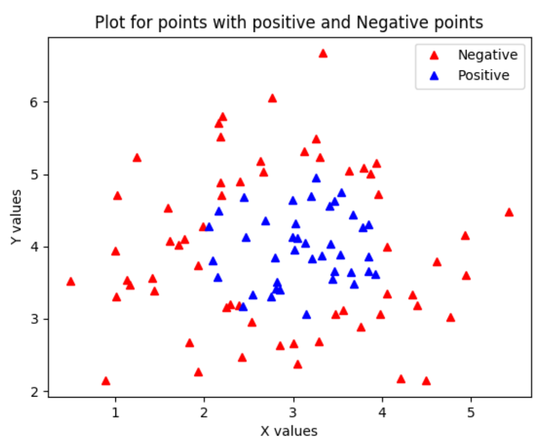
1.5.1 Observations :

- Since is the data in higher dimension(>2), we are plotting predicted values to understand decision boundary of model.
- To get approximate decision boundary of model, we can draw convex hull of predicted positive labels.

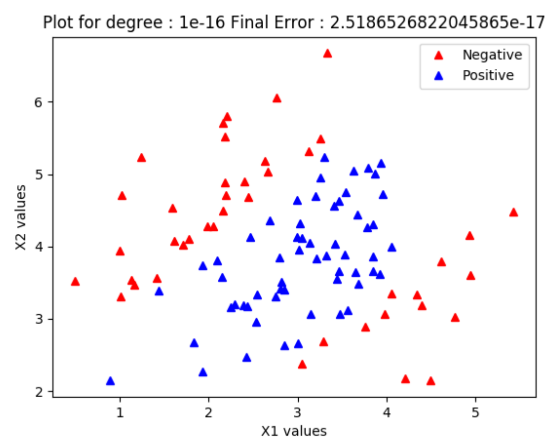
1.6 Effect of Regularisation parameter on Decision Boundary :

Now we perform feature transformation to increase the number of attributes and perform regularised logistic Regression. For Different Regularisation parameters, we calculate the error values along with that we plot prediction of new models for each regularisation parameter i.e.., lambda.

For Lambda = 1e-16

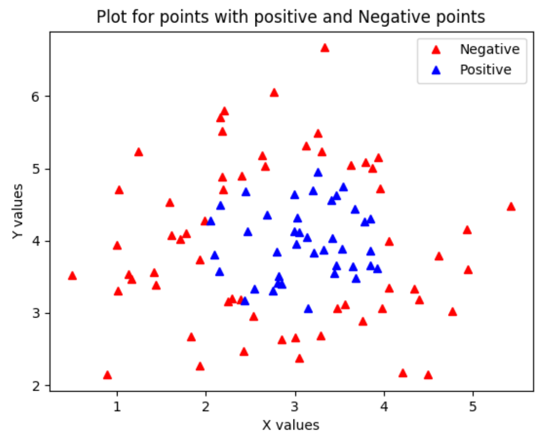


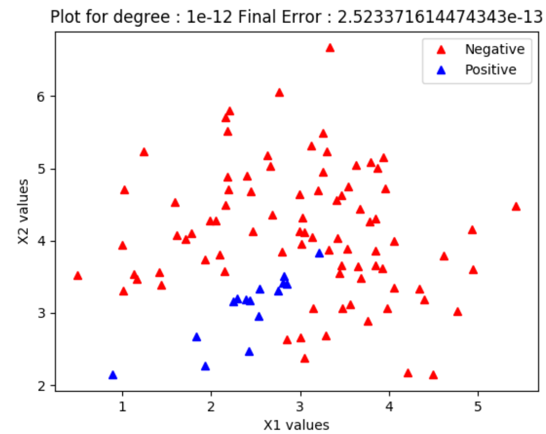
Actual values



Model prediction with Lambda = 1e-16

For Lambda = 1e-12

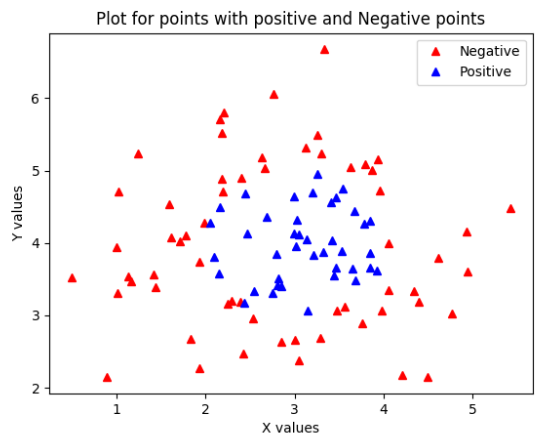


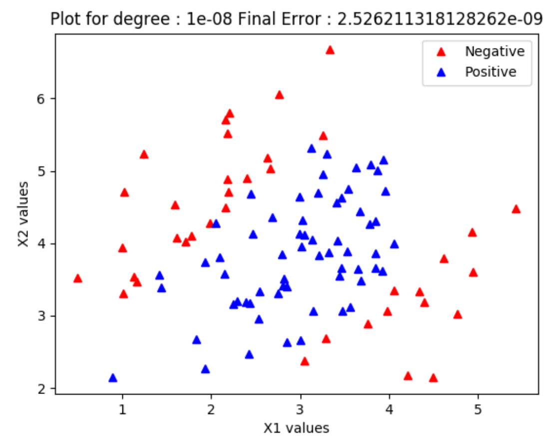


Actual values

Model prediction with Lambda = 1e-12

For Lambda = 1e-8

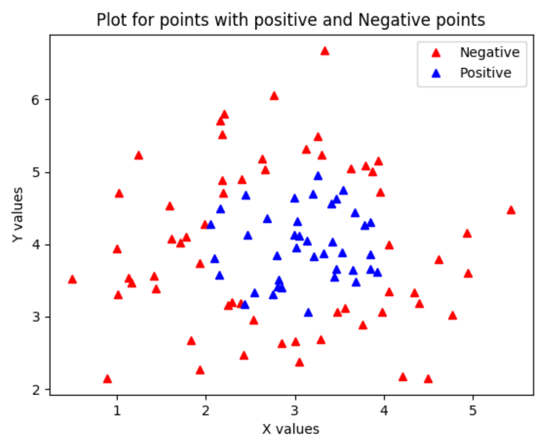


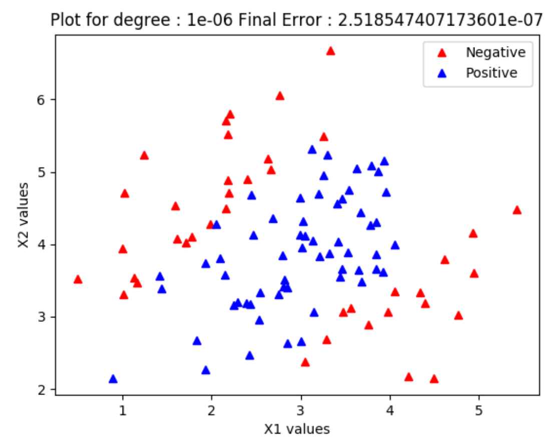


Actual values

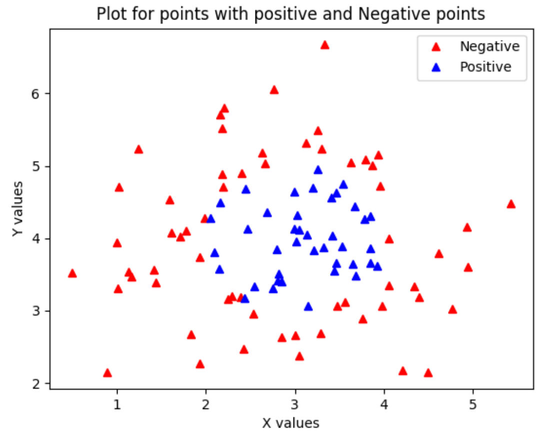
Model prediction with Lambda = 1e-8

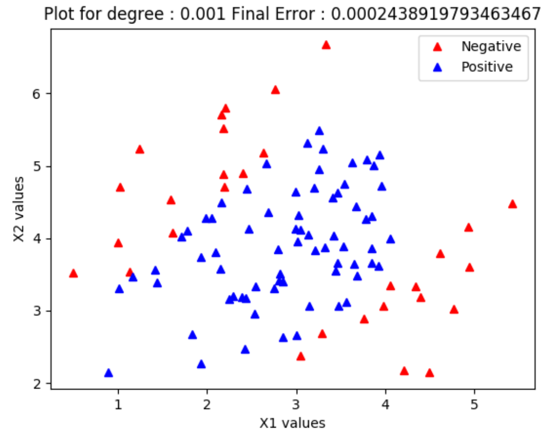
For Lambda = 1e-6





For Lambda = 0.001

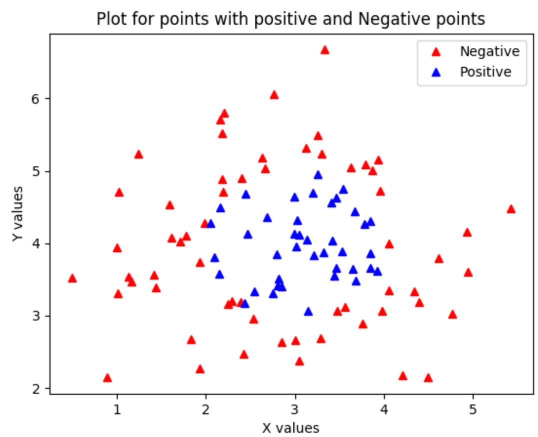


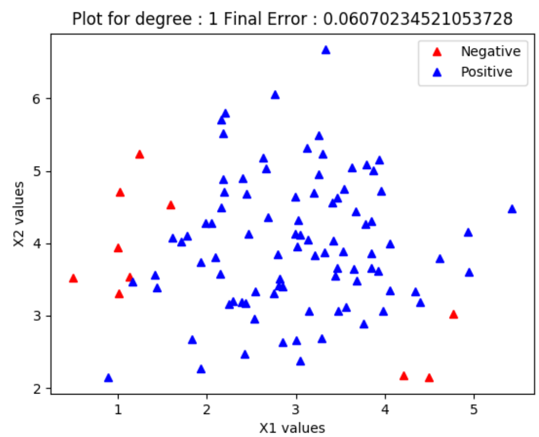


Actual values

Model prediction with Lambda = 0.001

For Lambda = 1

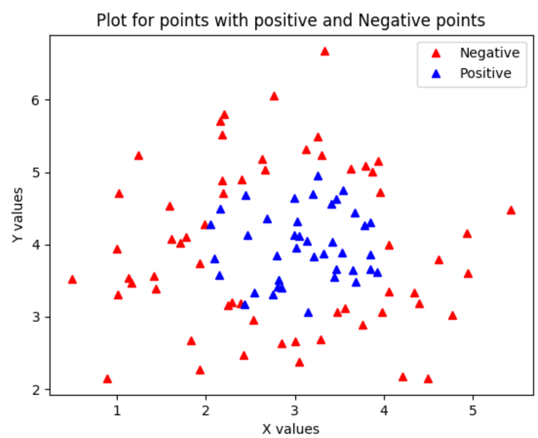


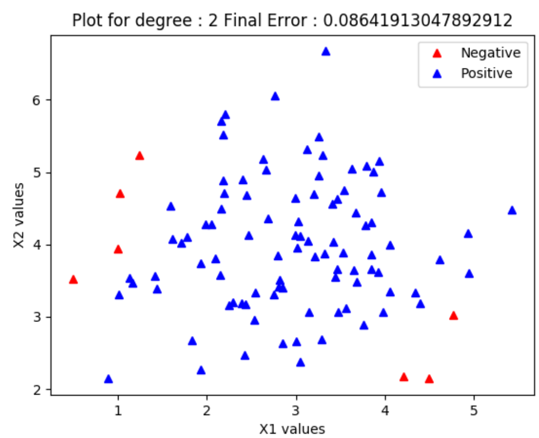


Actual values

Model prediction with Lambda = 1

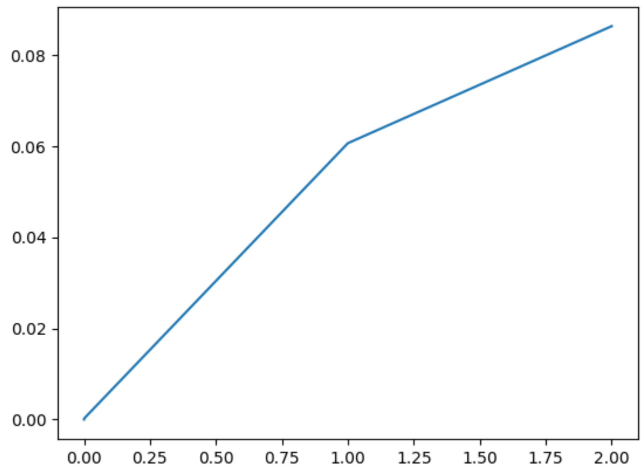
For Lambda = 2





1.6.1 Error Vs Regularisation parameter Graph :

As observed above predictions of model for various regularisation parameters. Now we plot Error Vs Regularisation parameters to understand the model performance with respect to Regularisation parameter.



Cross Entropy error Vs Regularisation parameter Values

1.6.2 Observations :

- It is observed that for small Lambda values, we are getting minimum error. In this case such Lambda = 1e -16.
- As lambda Values Increases, cross Entropy Error values increases and gets saturated afterwards.
- It is important task to tune lambda value such that it complements model performance.